Preprint typeset in JHEP style - HYPER VERSION ZU-TH 06/10

Integrated dipoles with MadDipole in the MadGraph framework

R. Frederix[a] **T. Gehrmann**[a], **N. Greiner**[a,b]

[a] *Institut für Theoretische Physik, Universität Zürich, Winterthurerstrasse 190, 8057 Zürich, Switzerland*

[b] *Department of Physics, University of Illinois at Urbana-Champaign, 1110 West Green Street, Urbana, IL 61801, USA*

Abstract: Heading towards a full automation of next-to-leading order (NLO) QCD corrections, one important ingredient is the analytical integration over the one-particle phase space of the unresolved particle that is necessary when adding the subtraction terms to the virtual corrections. We present the implementation of these integrated dipoles in the MadGraph framework. The result is a package that allows an automated calculation for the NLO real emission parts of an arbitrary process.

Contents

1. Introduction

Multi-particle final states are the basis of many physics studies at the CERN LHC. In searching for physics beyond the standard model, one is aiming to identify new particles from their decay products, which could often result from decay chains. Likewise, accompanying final state particles may help to improve the ratio of signal to background processes, as done for example in the Higgs boson search through the vector boson fusion channel. Meaningful searches for these signatures require not only a very good anticipation of the expected signal, but also of all standard model background processes which could result in identical final state signatures. From the theoretical point of view, high precision implies

that one has to go beyond the leading order in perturbation theory to be able to keep up with the precision of the measurements.

For leading order processes there have been many developments concerning event generation and simulation tools in the last two decades such as MadGraph/MadEvent [1–3] CompHEP/CalcHEP [4]/ [5], SHERPA [6, 7] and WHIZARD [8] and also programs using different approaches such as ALPGEN [9] and HELAC [10]. All these programs are multipurpose event generator tools, which are able to compute any process (up to technical restrictions in the multiplicity) within the standard model, or within alternative theories specified by their interaction Lagrangian or Feynman rules. They usually provide event information which can be passed into parton shower, hadronization and/or detector simulation through standard interfaces.

Next-to-leading order (NLO) calculations are at present performed on a process-by-process basis. The widely-used programs MCFM [11, 12], NLOJET++ [13], MC@NLO [14–17] and programs based on the POWHEG method [18–23] collect a variety of different processes in a standardized framework, the latter two methods also match the NLO calculation onto a parton shower. The POWHEG box [24] provides a toolkit for adapting further NLO calculations to match onto parton showers.

The NLO QCD corrections to a given process with a n-parton final state receive two types of contributions: the one-loop virtual correction to the $(2 \to n)$-parton scattering process, and the real emission correction from all possible $(2 \to n+1)$-parton scattering processes. For the numerical evaluation, one has to be able to compute both types of contributions separately.

The computation of one-loop corrections to multi-particle scattering amplitudes was performed on a case-by-case basis up to now, the calculational complexity increased considerably with increasing number of external partons. Since only a limited number of one-loop integrals can appear in the final result [25–27], the calculation of one-loop corrections can be reformulated as the determination of the coefficients of these basis integrals, plus potential rational terms. Based on this observation, a variety of methods for the systematic determination of the one-loop integral coefficients and the rational terms have been formulated, and first fully automated programs for the calculation of one-loop multi-parton amplitudes are becoming available with the packages CutTools [28, 29], BlackHat [30], Rocket [31] and GOLEM [32], as well as independent libraries [33].

These recent technical advances have allowed the calculation of NLO corrections to several $2 \to 4$ processes, which is the current frontier in complexity. The first result in this class of processes were the electroweak corrections to four-fermion production in electron-positron annihilation [34], based on [33]. Most recently NLO QCD corrections were obtained for genuine $2 \to 4$ processes at hadron colliders: $W+3$ jet production [35–38], $Z+3$ jet production [39], $pp \to t\bar{t}b\bar{b}$ [40–42] and $pp \to t\bar{t}jj$ [43], as well as for the quark-antiquark contribution to $pp \to b\bar{b}b\bar{b}$ [44].

The real emission corrections contain soft and collinear singularities, which become explicit only after integration over the appropriate real radiation phase space yielding a hard n-parton final state. They are canceled by the singularities from the virtual one-loop contributions, thus yielding a finite NLO correction. To systematically extract the

real radiation singularities from arbitrary processes, a variety of methods, based either on phase-space slicing [45] or on the introduction of process-independent subtraction terms [46] have been proposed. Several different algorithms to derive subtraction terms are available: residue (or FKS) subtraction [47] and variants thereof [48], dipole subtraction [49, 50] and antenna subtraction [51–54].

Especially the dipole subtraction formalism, which provides local subtraction terms for all possible initial and final state configurations [49] and allows to account for radiation off massive partons [50], is used very widely in NLO calculations. The generation of dipole terms for subtracting the singular behaviour from the real radiation subprocesses has been automated in various event generators: in the SHERPA framework [55], the TeVJet framework [56], the HELAC framework [57] and in the form of independent libraries [58] interfaced to MadGraph. The MadDipole package [59] provides an implementation within MadGraph. An implementation of the residue subtraction method is also available within MadGraph [60].

For a full NLO calculation, the dipole terms have to be integrated over the dipole phase space, and added with the virtual corrections to obtain the cancellation of infrared singularities. So far, only one of the implementations [57] provides these integrated dipole terms including all masses and possible phase-space restrictions, and constructs the corresponding integrated subtraction terms. It is the purpose of this paper to implement the generation of the integrated subtraction terms into MadDipole, which will consequently allow to carry out the full dipole subtraction within the MadGraph/MadEvent framework. The output results are `Fortran` subroutines which return the squared amplitude for all possible unintegrated and integrated dipole configurations in the usual MadGraph style.

With a complete treatment of the real NLO radiation, MadDipole is a crucial building block of automated NLO calculations. This automation is a very high priority for LHC predictions [61], and can likely be accomplished only in a collaborative effort, with different groups supplying different pieces, linked through standard interfaces [62].

The paper is structured as follows: in Section 2, we briefly review the structure of unintegrated and integrated dipole terms, Section 3 discusses the expansion of the integrated dipoles, and normalization conventions used in this. The MadDipole implementation of the integrated dipoles is described in Section 4, with functionality checks documented in Section 5. Finally, we conclude with Section 6.

2. Structure of NLO dipole subtraction terms

The fundamental building blocks of the subtraction terms in the dipole formalism [49, 50] are dipole splitting functions $\mathbf{V}_{ij,k}$, which involve only three partons: emitter i, unresolved parton j, spectator k. A dipole splitting function accounts for the collinear limit of j with i, and for part of the soft limit of j in between i and k. The dipole factors, which constitute the subtraction terms, are obtained by multiplication with reduced matrix elements, where partons i, j and k are replaced by recombined pseudo-partons $\widetilde{ij}$, $\widetilde{k}$. The full soft behavior is recovered after summing all dipole factors.

Throughout the whole paper we are using the notation introduced in Refs. [49] and [50]. The dipole factors are subtracted from the real radiation contribution at NLO. They subtract the singular contributions where one parton from the NLO real radiation contribution becomes soft and/or collinear, such that the phase-space integral of this contribution can be preformed numerically, including arbitrary kinematic restrictions on the final-state phase space.

The dipole factors are integrated analytically over the dipole phase space (which fully includes the infrared singular regions), such that the integrated dipole factors have the same kinematic structure as the virtual one-loop NLO corrections and the collinear counterterms from mass factorization. These integrated dipole contributions can then be added to the virtual and mass-factorization corrections, thereby accomplishing the cancellation of infrared singularities.

Several algorithms to automatically generate the unintegrated dipole terms for arbitrary processes have been devised and implemented in various matrix element generator frameworks. They are available for SHERPA [55], HELAC [57], for MadGraph [59] as well as a implementation of stand-alone routines [56, 58]. Also implementations of integrated dipoles are available in the same codes. However, only the implementation based on the HELAC framework has the full set of integrated dipoles for arbitrarly masses and phase-space restrictions [57]. It was used in the context of the calculations of NLO corrections to $pp \to t\bar{t}b\bar{b}$ [42] and $pp \to t\bar{t}jj$ [43]. In this work, we document our implementation of the integrated dipoles in MadDipole, which is a package in the MadGraph framework. With this extension, MadDipole computes the full NLO dipole subtraction of real radiation and performs the infrared cancellations in a fully automated manner for color- and helicity-summed matrix elements squared. Our implementation was already used in the computation of the NLO corrections to $pp \to b\bar{b}b\bar{b}$ in the quark-initiated channel [44].

2.1 Phase space restriction: α-parameter

The calculation of the subtraction terms is only necessary in the vicinity of a soft and/or collinear limit. Away from these limits the amplitude is finite and there is in principle no need to calculate the computationally heavy subtraction terms. The distinction between regions near to a singularity and regions without need for a subtraction can be parametrized by a parameter usually labelled α with $\alpha \in [0, 1]$, which was introduced in Ref. [63] for processes involving partons only in the final state. The case with incoming hadrons, *i.e.*, with partons in the initial state, is described in Ref. [13].

Using the notation of Ref. [13], the contribution from the subtraction term to the

differential cross section in the real radiation channel can be written as

$$
\begin{aligned}
d\sigma^A_{ab} = & \sum_{\{n+1\}} d\Gamma^{(n+1)}(p_a,p_b,p_1,...,p_n+1)\frac{1}{S_{\{n+1\}}} \\
\times & \Bigg\{ \sum_{\substack{\text{pairs}\\ i,j}} \sum_{k\neq i,j} \mathcal{D}_{ij,k}(p_a,p_b,p_1,...,p_{n+1}) F_J^{(n)}(p_a,p_b,p_1,..,\tilde{p}_{ij},\tilde{p}_k,..,p_{n+1})\Theta(y_{ij,k}<\alpha) \\
& + \sum_{\substack{\text{pairs}\\ i,j}} \Big[\mathcal{D}^a_{ij}(p_a,p_b,p_1,...,p_{n+1}) F_J^{(n)}(\tilde{p}_a,p_b,p_1,..,\tilde{p}_{ij},..,p_{n+1})\Theta(1-x_{ij,a}<\alpha) \\
& \qquad + (a\leftrightarrow b)\Big] \\
& + \sum_{i\neq k} \Big[\mathcal{D}^{ai}_{k}(p_a,p_b,p_1,...,p_{n+1}) F_J^{(n)}(\tilde{p}_a,p_b,p_1,..,\tilde{p}_k,..,p_{n+1})\Theta(u_i<\alpha) + (a\leftrightarrow b)\Big] \\
& + \sum_{i} \Big[\mathcal{D}^{ai,b}(p_a,p_b,p_1,...,p_{n+1}) F_J^{(n)}(\tilde{p}_a,p_b,\tilde{p}_1,...,\tilde{p}_{n+1})\Theta(\tilde{v}_i<\alpha) + (a\leftrightarrow b)\Big] \Bigg\} \ .
\end{aligned}
\tag{2.1}
$$

The functions $\mathcal{D}_{ij,k}$, $\mathcal{D}^a_{ij}$, $\mathcal{D}^{ai}_k$ and $\mathcal{D}^{ai,b}$ are the dipole terms for the various combinations for emitter and spectator. $\sum_{\{n+1\}}$ denotes the summation over all possible configurations for this $(n+1)$-particle phase space which is labelled as $d\Gamma^{(n+1)}$ and the factor $S_{\{n+1\}}$ is the symmetry factor for identical particles. In MadDipole, we have introduced four different α-parameters, one for each type of dipoles [59]. In our code they are called `alpha_ff`, `alpha_fi`, `alpha_if` and `alpha_ii` for the final-final, finial-initial, initial-final and initial-initial dipoles, respectively. The actual values for these parameters are by default set to unity, corresponding to the original formulation of the dipole subtraction method [49, 50], but can be changed by the user.

The integrated dipole factors, which are to be added with the virtual n-parton contribution, also depend on α. For case of massless partons, the α-dependence of the integrated terms is stated in [13, 63] while for massive partons results for most cases can be found in [57, 64, 65]. The remaining cases, *i.e.*, the (finite) massless-to-massive splittings, can be found in the appendix.

2.2 Regularization scheme dependence

Calculating objects that contain divergences requires a systematic prescription of how to deal with these divergences, *i.e.*, a scheme for their regularization. In NLO calculations, the same regularization scheme has to be applied in the real emission part and in the virtual corrections. Both these contributions will differ between different regularization schemes, while their sum (*i.e.*, the full NLO result) is scheme-independent. Therefore it is necessary to clearly specify which regularization scheme one is using.

In QCD calculations, there are mainly two types of regularization schemes used, namely dimensional regularization [66–69] and dimensional reduction [70–72]. Both extend the

dimensionality of space-time to $d = 4 - 2\epsilon$, resulting in divergences becoming explicit as poles in $1/\epsilon$. A discussion about their subtypes and their differences can be found in [73].

In the real radiation contribution, the dependence on the regularization scheme does not yet appear explicitly at the level of the unintegrated dipole terms, and we consequently did not address this issue in the previous release of MadDipole [59].

The helicity subroutines on which MadGraph and MadDipole are build evaluate matrix elements in four dimensions. Therefore we can compute the subtraction terms only in regularization schemes in which the external particles are defined in four dimensions. The two most widely used are the 't Hooft-Veltman scheme (tHV) in dimensional regularization, which is the default used in our implementation, and the four-dimensional helicity scheme (FDH). Both methods differ only by a finite shift [49, 74]:

$$\mathcal{V}_I(\epsilon)^{\text{tHV}} \to \mathcal{V}_I^{\text{FDH}}(\epsilon) = \mathcal{V}_I(\epsilon)^{\text{tHV}} - \tilde{\gamma}_I + \mathcal{O}(\epsilon) \ , \tag{2.2}$$

$$\tilde{\gamma}_q = \tilde{\gamma}_{\bar{q}} = \frac{1}{2} C_F \ , \qquad \tilde{\gamma}_g = \frac{1}{6} C_A \ . \tag{2.3}$$

In the massive case there is no dependence on the regularization scheme [75]. There is a simple flag in our code that allows to change between these two schemes.

3. Expansion of integrated dipoles in ϵ

Integration of the dipoles makes their infrared singularities explicit as poles in the dimensional regularization parameter ϵ. The formal structure of the integrated dipoles is independent of the configuration we are considering. For definiteness, we discuss only the final-final case, the same structure also holds in all other cases. For initial state hadrons, a additional collinear contribution is present, which is rendered finite by mass factorization, which we will describe here as well.

3.1 Infrared final-state singularities

In the final-final case, the integrated dipole function is written as

$$\int [\mathrm{d}p_i(\widetilde{p}_{ij}, \widetilde{p}_k)] \, \frac{1}{(p_i + p_j)^2 - m_{ij}^2} \, \langle \mathbf{V}_{ij,k} \rangle \ \equiv \ \frac{\alpha_\mathrm{s}}{2\pi} \frac{1}{\Gamma(1-\epsilon)} \left(\frac{4\pi\mu^2}{2\widetilde{p}_{ij} \cdot \widetilde{p}_k} \right)^\epsilon \mathcal{V}_{ij}(\epsilon) \, . \tag{3.1}$$

Depending on the configuration, the splitting function and the propagator on the left hand side of (3.1) change their form; the structure of the result on the right hand side remains the same.

The factor $\mathcal{V}_{ij}(\epsilon)$ is determined by the specific configuration. It is singular in the limit $\epsilon \to 0$, and is expanded as a Laurent series in ϵ. Prefactors taken out from the expansion must be consistent between the dipoles and the virtual one-loop corrections to the process under consideration.

In the $\overline{\text{MS}}$-scheme, only universal factors are taken out, and this expansion can be written symbolically as

$$\begin{aligned}\int [\mathrm{d}p_i(\widetilde{p}_{ij},\widetilde{p}_k)]\,\frac{1}{(p_i+p_j)^2-m_{ij}^2}\,\langle \mathbf{V}_{ij,k}\rangle \;&\equiv\; \frac{\alpha_\mathrm{s}}{2\pi}\frac{1}{\Gamma(1-\epsilon)}\left(\frac{4\pi\mu^2}{2\widetilde{p}_{ij}\cdot\widetilde{p}_k}\right)^{\epsilon}\mathcal{V}_{ij}(\epsilon)\\ &= \frac{e^{\gamma\epsilon}}{(4\pi)^\epsilon}\left(\frac{y_{ij;1}}{\epsilon^2}+\frac{y_{ij;2}}{\epsilon}+y_{ij;3}\right),\end{aligned} \tag{3.2}$$

where γ is the Euler-Mascheroni constant, $\gamma = 0.5772\ldots$. This structure is the basis for our implementation.

The specific values of $y_{ij;1}$, $y_{ij;2}$ and $y_{ij;3}$ depend on the splitting one is considering. For instance if one takes the massless final-final quark-gluon splitting, *i.e.*, where the emitter is a massless quark int the final state, the unresolved particle is a gluon, and the spectator is also a massless final state particle, then the integrated splitting function is given by

$$\mathcal{V}_{qg}(\epsilon) = C_\mathrm{F}\left(\frac{1}{\epsilon^2}+\frac{3}{2\epsilon}+5-\frac{\pi^2}{2}\right). \tag{3.3}$$

Consequently, the expansion coefficients in (3.2) become

$$\begin{aligned} y_{qg;1} &= \frac{\alpha_\mathrm{s}}{2\pi}C_\mathrm{F},\\ y_{qg;2} &= \frac{\alpha_\mathrm{s}}{2\pi}C_\mathrm{F}\left(\frac{3}{2}+\log\left(\frac{\mu^2}{2\widetilde{p}_{ij}\cdot\widetilde{p}_k}\right)\right),\\ y_{qg;3} &= \frac{\alpha_\mathrm{s}}{2\pi}C_\mathrm{F}\left(\frac{1}{2}\log^2\left(\frac{\mu^2}{2\widetilde{p}_{ij}\cdot\widetilde{p}_k}\right)+\frac{3}{2}\log\left(\frac{\mu^2}{2\widetilde{p}_{ij}\cdot\widetilde{p}_k}\right)-\frac{7\pi^2}{12}+5\right). \end{aligned} \tag{3.4}$$

3.2 Initial-state collinear behaviour

The cases with initial state radiation, *i.e.*, initial-initial and initial-final, are slightly more involved. Not all singularities that occur in the real emission process are cancelled by the virtual corrections. The integrated initial-final dipole functions take the form:

$$\begin{aligned}&\int [\mathrm{d}p_i(\widetilde{p}_k;p_a,z)]\,\frac{1}{(p_i+p_j)^2-m_{ij}^2}\,\frac{n_s(\widetilde{ai})}{n_s(a)}\,\langle \mathbf{V}_k^{ai}\rangle\\ &\equiv \frac{\alpha_\mathrm{s}}{2\pi}\frac{1}{\Gamma(1-\epsilon)}\left(\frac{4\pi\mu^2}{2p_a\cdot\widetilde{p}_k}\right)^{\epsilon}\mathcal{V}^{a,ai}(z;\epsilon)\\ &= \frac{e^{\gamma\epsilon}}{(4\pi)^\epsilon}\left(\frac{y_{i,j;1}(z)}{\epsilon^2}+\frac{y_{i,j;2}(z)}{\epsilon}+y_{i,j;3}(z)\right).\end{aligned} \tag{3.5}$$

The left-over singularities arise from collinear splitting off the initial emitter particle. For example, for the initial-final dipole describing the gluon radiation off an incoming quark, one has

$$\mathcal{V}^{q,q}(z;\epsilon) = -\frac{1}{\epsilon}P^{qq}(z)+\delta(1-z)\left[\mathcal{V}_{qg}(\epsilon)+\left(\frac{2\pi^2}{3}-5\right)C_\mathrm{F}\right]+B^{q,q}(z)+\mathcal{O}(\epsilon), \tag{3.6}$$

where $B^{q,q}(z)$ contains regular functions and plus-distributions in z.

The collinear singularity remains and is absorbed into the parton distribution function. This is done by introducing a collinear counterterm and its contribution to the cross section is given by (6.6) of [49]:

$$d\sigma_a^C(p;\mu_F^2) = -\frac{\alpha_s}{2\pi}\,\frac{1}{\Gamma(1-\epsilon)}\sum_b \int_0^1 dz\,\left[-\frac{1}{\epsilon}\left(\frac{4\pi\mu^2}{\mu_F^2}\right)^\epsilon P^{ab}(z) + K^{ab}_{\overline{\text{F.S.}}}(z)\right] d\sigma_b^B(zp)\;, \quad (3.7)$$

where in $\overline{\text{MS}}$ scheme, $K^{ab}_{\overline{\text{F.S.}}}(z) = 0$.

Neglecting the sum over the partonic subprocesses, we have a counterterm contribution of the form

$$I_c^{ab}(\epsilon) = -\frac{\alpha_s}{2\pi}\frac{1}{\Gamma(1-\epsilon)}\left(\frac{4\pi\mu^2}{\mu_F^2}\right)^\epsilon\left[-\frac{1}{\epsilon}P^{ab}(z)\right] \quad (3.8)$$

which is added to the integrated dipole.

In the $\overline{\text{MS}}$ scheme the expansion of (3.8) is given by

$$I_c^{ab}(\epsilon) = \frac{e^{\gamma\epsilon}}{(4\pi)^\epsilon}\left(\frac{y^c_{a,b;2}}{\epsilon} + y^c_{a,b;3}\right)\;. \quad (3.9)$$

The coefficients $y^c_{a,b;2}$ and $y^c_{a,b;3}$ are given by

$$\begin{aligned} y^c_{a,b;2} &= \frac{\alpha_s}{2\pi}\cdot P^{ab}(z) \\ y^c_{a,b;3} &= \frac{\alpha_s}{2\pi}\cdot P^{ab}(z)\log\left(\frac{\mu^2}{\mu_F^2}\right)\;. \end{aligned} \quad (3.10)$$

3.3 Structure of program output

The final output the program provided to the user is then given by all contributions of the coefficients y_i and y_i^c respectively multiplied with a born level matrix element that is modified by its color structure. This explicitly means

$$\begin{aligned} &\tfrac{1}{\epsilon^2}\text{:} && y_1\cdot\ {}_m\langle 1\cdots m|\frac{\boldsymbol{T}_i\cdot\boldsymbol{T}_k}{\boldsymbol{T}_i^2}|1\cdots m\rangle_m\;, \\ &\tfrac{1}{\epsilon}\text{:} && (y_2+y_2^c)\cdot\ {}_m\langle 1\cdots m|\frac{\boldsymbol{T}_i\cdot\boldsymbol{T}_k}{\boldsymbol{T}_i^2}|1\cdots m\rangle_m\;, \\ &\text{finite:} && (y_3+y_3^c)\cdot\ {}_m\langle 1\cdots m|\frac{\boldsymbol{T}_i\cdot\boldsymbol{T}_k}{\boldsymbol{T}_i^2}|1\cdots m\rangle_m\;. \end{aligned}$$

The contributions from the collinear counterterms are of course only present if there are initial state QCD particles involved. By adding the collinear counterterm, only endpoint singularities, which occur at $z = 1$, remain. Those then completely cancel with the virtual corrections. The finite pieces can contain regular functions of z as well as δ-functions and plus-distributions.

4. Implementation and how to use the integrated dipoles

The installation and running of the new package is very similar to the already existing MadDipole package:

1. Download the MadDipole package (version 4.4.35 or later), `MG_ME_DIP_V4.4.??.tar.gz`, from one of the MadGraph websites, *e.g.*, `http://madgraph.phys.ucl.ac.be/`.

2. Extract and run `make` in the `MadGraphII` directory.

3. Copy the `Template` directory into a new directory, *e.g.*, `MyProcDir` to ensure that you always have a clean copy of the Template directory.

4. Go to the new `MyProcDir` directory and specify your process in the file `./Cards/proc_card.dat`. This is the $(n+1)$-particle process you require the subtraction term for.

5. Running `./bin/newprocess` generates the code for the $(n+1)$-particle matrix element and for all dipole terms and their integrated versions. After running this you will find a newly generated directory `./SubProcesses/P0_yourprocess` (*e.g.*, `./SubProcesses/P0_e+e-_uuxg`) which contains all required files.

In the `./SubProcesses/P0_yourprocess` directory all the files relevant to that particular subprocesses are generated. In particular this includes the $(n+1)$ particle matrix elements in the file `matrix.f` and the dipoles in the files `dipol???.f`, where `???` stands for a number starting from `001`. Furthermore the directory has two files, `dipolsum.f` and `intdipoles.f`, where the sum of the dipoles and their integrated versions are calculated, respectively.

Here we discuss in more detail the syntax and implementation of the integrated dipoles in the `intdipoles.f` file. In this file there are two subroutines of the form

```
intdipoles(P, X, Z, PSWGT EPSSQ, EPS, FIN) and
intdipolesfinite(P, X, Z, PSWGT EPSSQ, EPS, FIN),
```

where the input parameters are the phase space point `P(0:3,nexternal)`, the Bjorken x values of both incoming parton distributions, `X(1:2)`, and the momentum fraction of the incoming parton that goes into the hard process after an initial state radiation or if the spectator is an initial state particle, `Z`. It is the latter quantity which is denoted with $x_{ij,a}$, $x_{ik,a}$ and $x_{i,ab}$ respectively in [49], but we shall refer to it here simply as z. In the numerical integration, it must be taken uniformly over the interval $z \in [0;1]$ to ensure correct representation of the distributions. Furthermore, the phase space weight should be passed as well. These first four arguments should be provided by the user. For the given phase space point, the routines evaluate the sum of all integrated dipole subtraction terms (integrated dipole factors multiplied with the appropriate reduced matrix elements) after mass factorization of the collinear singularities. The routines call external subroutines (explained below) from `dipolesum.f` which supply the parton distributions appearing with the reduced matrix elements, apply event rejection cuts and pass the event information into histograms.

The integrated dipoles in the two subroutines correspond to the unintegrated dipoles in `dipolsum(..)` and `dipolsumfinite(..)`. The dipoles in the `dipolsumfinite(..)` and `intdipolesfinite(..)` subroutines are not needed to cancel singularities because they correspond to gluon splittings into massive particles. However, they can be useful for

checks when taking the limit of vanishing quark masses, or to cancel some large logarithms in the real emission matrix elements [59].

The output parameters are the coefficients of the divergent and finite terms: `EPSSQ` is the coefficient of $1/\epsilon^2$, `EPS` is the coefficient of $1/\epsilon$. After inclusion of the collinear counterterms they contribute with a factor $\delta(1-z)$, and are real numbers. `FIN` is the finite coefficient. The calculation of the `FIN` coefficient requires some explanation, since its coming from distributions in z. In the `intdipoles(..)` and `intdipolesfinite(..)` subroutines a three-dimensional array `FINITE` is filled with the contributions from the various dipoles:

`FINITE(1)`: regular function in z,
`FINITE(2)`: coefficient of $\delta(1-z)$,
`FINITE(3)`: coefficient of $\delta(z_+-z)$.

The last entry appears only for massive dipoles, with

$$z_+ = 1 - 4\mu_Q^2, \tag{4.1}$$

where μ_Q is the rescaled fermion mass occurring in the splitting.

In this decomposition, the δ-functions and (+)-distributions are carried out by taking into account the convolution with the product of a reduced matrix element $g(z)$ (generated by MadDipole out of MadGraph) and a parton distribution function $h(z)$ (supplied by the user through the subroutine `dipolepdf`)

$$\int_0^1 \mathrm{d}z\,\delta(1-z)g(z)h(z) = \int_0^1 \mathrm{d}z\ \underbrace{g(1)\,h(1)}_{\text{in FINITE(2)}}\,, \tag{4.2}$$

$$\int_0^1 \mathrm{d}z\,(f(z))_+\,g(z)\,h(z) = \int_0^1 \mathrm{d}z\Big[\underbrace{f(z)\,g(z)\,h(z)}_{\text{in FINITE(1)}} - \underbrace{f(z)\,g(1)\,h(1)}_{\text{in FINITE(2)}}\Big]. \tag{4.3}$$

In the massive case the point $z=1$ can not be reached in all cases but we may have a reduced endpoint z_+, such that instead of a $\delta(1-z)$ we then have a δ-distribution of the form $\delta(z_+-z)$ which is the third entry of the array of the finite terms. If we have such a reduced endpoint then also the (+)-distribution is generalized into a z_+-distribution as defined in (A.13). As before, we have the following implementation:

$$\int_0^1 \mathrm{d}z\,\delta(1-z_+)g(z)h(z) = \int_0^1 \mathrm{d}z\ \underbrace{g(z_+)\,h(z_+)}_{\text{in FINITE(3)}}, \tag{4.4}$$

$$\int_0^1 \mathrm{d}z\,(f(z))_{z_+}\,g(z)\,h(z) = \int_0^1 \mathrm{d}z\ \Theta(z_+-z)\Big[\underbrace{f(z)\,g(z)\,h(z)}_{\text{in FINITE(1)}} - \underbrace{f(z)\,g(z_+)\,h(z_+)}_{\text{in FINITE(3)}}\Big]. \tag{4.5}$$

By making a transformation of variables and computing the parton distribution function not at x (Bjorken's x) but at x/z, the matrix element itself becomes independent of z as the initial state particle, that radiates the unresolved particle, then comes with the momentum fraction of $x/z\cdot z = x$. Therefore the set of momenta which are used to

calculate the matrix element do not depend on z. This means that $g(z) = g(z_+) = g(1)$ in Eqs. (4.2–4.5).

The final result `FIN` is the sum of the three contributions:

$$\texttt{FIN} = \left(\texttt{FINITE(1)}/z + \texttt{FINITE(2)} + \texttt{FINITE(3)}/z_+\right) \times \texttt{PSWGT}. \tag{4.6}$$

Since `FINITE(3)` appears only in massless-to-massive splittings, it is always zero for the `intdipoles(..)` subroutine. Conversely, the `EPSSQ` and `EPS` should be zero when coming from the subroutine `intdipolesfinite(..)`.

For ease of use, we have provided three dummy subroutines/functions that the user might want to fill when using the code. In the code there are consistent calls to these subroutines. These subroutines/functions can be found in the file `dipolsum.f` and are

`passcutsdip(P):` In this `LOGICAL FUNCTION` the user should provide a set of cuts that he wants to be applied to the phase space points. Note that in general the phase-space mapping for each unintegrated dipole is different; there needs to be a call to this function for each dipole. It should return `FALSE` if the point fails the cuts. By default every point passes the cuts.

`dipolepdf(P,leg1,leg2,WGT):` In this `SUBROUTINE` the user should provide the value for his/her favourite PDF set for the two incoming particles with PDG codes passed by `leg1` and `leg2`. The factorization scale should be defined in the include file `dipole.inc`. The weight from the PDF should be returned in the argument `WGT`, which is set to `1` by default.

`writehist(P,WGT):` In this `SUBROUTINE` the phase-space point (for each dipole) are provided together with its weight. The user could use these to fill histograms or save ntuples.

Besides these three subroutines/functions, the more technical parameters, like the α-parameter (see Sec. 2.1), the number of flavors, the renormalization scheme and the scales can be set in the include file `dipole.inc`. When changing any of the parameters in this file the code should be recompiled (after removing the object files) for these changes to take effect.

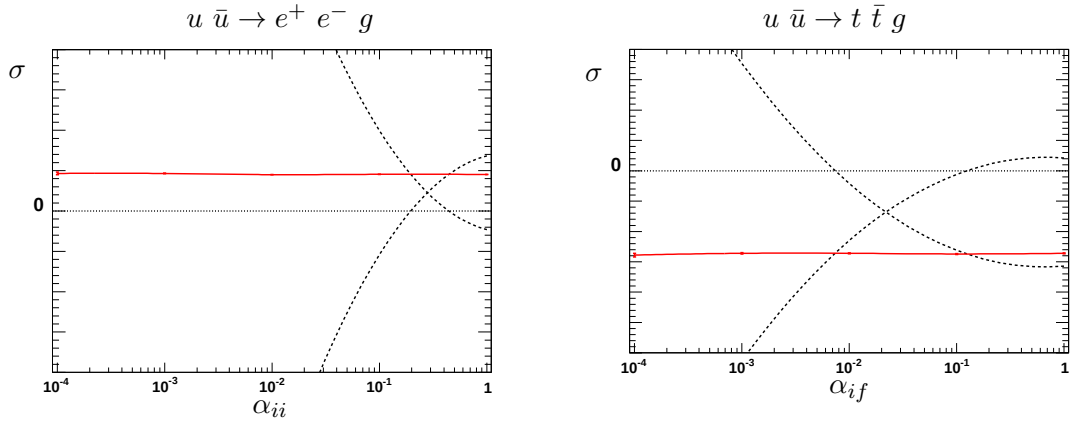
Besides the already existing `check` checking program, that checks the limits of the real emission matrix element minus the subtraction terms, we provide the user with another sample program, `checkint`, to calculate the value of the integrated subtraction terms for a given (or random) phase space point.

More details and latest news, updates, bug fixes, etc. can be found at

`http://cp3wks05.fynu.ucl.ac.be/twiki/bin/view/Software/MadDipole`.

5. Checks

To verify the implementation we have performed two different kinds of checks: independence on the phase space restriction parameter α and comparison with the implementation of dipoles in the MCFM program [11, 12].



(a) (b)

Figure 1: Two examples for the cancellation of the α-dependence between unintegrated and integrated dipoles on specific contributions. Since these do not correspond to full physical processes, we use an arbitrary normalization. The plot on the left shows the α dependence of the initial-initial dipole where a gluon is radiated of the initial quark lines. On the right the α dependence of an initial-final dipole where the spectator is massive is shown. For both figures only the α parameter shown is varied, all others are kept fixed. The upper dashed line is the first contribution of (5.1), *i.e.*, the sum of matrix element and unintegrated dipoles. The lower dashed line is the second contribution, *i.e.*, the finite terms of the integrated dipoles. The (red) solid line is the sum of both. For the sum we also included the Monte-Carlo error to show that the results for different values are consistent with each other.

Both the non-integrated and the integrated dipoles depend on the α-parameter, the dependence on this parameter should cancel in the sum. To validate this, we require:

$$\int_{n+1} \left(d\sigma^R - d\sigma^A\right) + \int_n (\text{finite parts of int. dip.}) = \text{const}, \tag{5.1}$$

which must be a constant in that sense that it should not depend on α. We have validated this for all 27 different cases (emitter/spectator in initial/final state, and mass assignments) listed in the appendix. Figure 1 shows the dependence on the α-parameter for two examples: in the left plot is the dependence shown on the α_{ii}-parameter that governs the initial-initial dipole phase-space restriction, and in the right plot for a processes with massive spectators the dependence on the α_{if}-parameter (that restricts the initial-final dipole phase space) is shown. The solid (red) lines display the quantity defined in (5.1), which is observed to be independent on α over several orders of magnitude.

This is a very powerful check because it includes several aspects of the implementation. It verifies not only the correct implementation of the θ-functions for the unintegrated subtraction terms and the α-dependent correction terms for the integrated dipoles, but independence on this parameter can only be achieved if the correct parton distribution functions are called with the right arguments, and if the various terms contributing to the integrated dipoles are summed correctly. Moreover, an inconsistent infrared-unsafe implementation of the cuts (by the user) will lead to a dependence on the α parameter.

The α-independence does, however, not probe important features related to the pole structure of the subtraction terms, namely the scheme dependence and the factorization

scale dependence, neither does it show other finite contributions. Therefore our second check was a direct comparison of our implementation against MCFM [11, 12]. We have compared the results for single phase space points, which allowed us to directly probe the implementation of the integrated terms. The various terms (regular functions, δ- and plus-distributions) could be checked separately. We note that not all possibilities (massless/massive, initial/final emitter/spectator) are present in MCFM and therefore not all could be checked. This includes in particular the finite dipoles: massless-to-massive splittings have no divergences, but do have a potentionally large logarithm which might be useful to subtract.

6. Conclusions

The MadDipole package [59] provides dipole subtraction terms for the evaluation of real radiation corrections in NLO calculations within the framework of the MadGraph/MadEvent matrix element and event generator [1–3]. In this work, we described the extension of the MadDipole package to include the integrated dipole terms, which are required to combine the real radiation corrections to a given process with the virtual corrections and collinear counterterms of the parton distributions. With the newly developed subroutines, MadDipole provides the unintegrated and integrated dipole subtraction terms.

The integrated subtraction terms are convoluted with the user-supplied parton distributions and are combined with the collinear counterterms from mass factorization. Consequently, infrared singularities appear only in the kinematic endpoints, which correspond to the kinematics of the virtual corrections. The MadDipole output can thus be readily combined with the results from one-loop matrix element generators, which are currently under rapid development [28–32].

A first application of the MadDipole package, in combination with the GOLEM one-loop amplitude generator [32], was in the calculation of NLO corrections to the quark-antiquark contribution to $pp \to b\bar{b}b\bar{b}$ [44]. Many more applications are likely to follow.

Acknowledgements

We would like to thank Fabio Maltoni for many useful discussions and John Campbell for some help on the checks against MCFM. NG wants to thank the University of Zurich for the kind hospitality during final work on the project. This research was supported in part by the Swiss National Science Foundation (SNF) under contract 200020-126691 and in part by the U. S. Department of Energy under contract No. DE-FG02-91ER40677.

A. α-dependence of the integrated dipoles

In this appendix we give a list of the references, where the α-dependent terms can be found in the literature. For the few cases which were not known previously, we give the result below.

A.1 Final-final

	Configuration	Reference	Remarks
(a)	$Q \to Q\ g,\ m_k > 0$	[42] (A.9,A.17,A.20)	Different definition of α
(b)	$Q \to Q\ g,\ m_k = 0$	[64] (A.22,A.23)	
(c)	$q \to q\ g,\ \ m_k > 0$	[64] (A.29,A.31)	
(d)	$q \to q\ g,\ \ m_k = 0$	[63] (22)	
(e)	$g \to g\ g,\ \ m_k > 0$	[42] (A.9,A.19,A.22)	Different definition of α
(f)	$g \to g\ g,\ \ m_k = 0$	[63] (24)	
(g)	$g \to Q\ \bar{Q},\ m_k > 0$	this work, (A.6)	
(h)	$g \to Q\ \bar{Q},\ m_k = 0$	this work, (A.9)	
(i)	$g \to q\ \bar{q},\ \ m_k > 0$	[42] (A.18,A.21)	Different definition of α
(j)	$g \to q\ \bar{q},\ \ m_k = 0$	[63] (23)	

In the massive case there is an ambiguity how one defines α. The upper limit of the integration over y is given by the variable y_+. One can now define α in such a way that the maximal value for α is given by y_+. In that case the integration range is given by the interval $y \in [\alpha, y_+]$. Another possibility is to have $\alpha = 1$ as the maximal value which implies that the allowed integration range is given by

$$\alpha y_+ \ < \ y_{ij,k} \ < \ y_+. \tag{A.1}$$

The first definition has been used in [42] whereas we use the latter.

The two cases (g) and (h) are in principle not needed as the result is still finite due to the masses of the quarks. For the sake of completeness we also included this cases and therefore give the result.

A.1.1 Case (g)

Case (g) is the splitting of a gluon into two massive quarks ($g \to Q\ \bar{Q}$) with a massive spectator. The splitting function for that process is given in Eq.(5.18) of [50],

$$\begin{aligned}\langle \mathbf{V}_{Q\bar{Q},k} \rangle = 8\pi\mu^{2\epsilon}\alpha_{\mathrm{s}} T_{\mathrm{R}} \frac{1}{v_{ij,k}} \Bigg\{ 1 - \frac{2}{1-\epsilon} \Bigg[\tilde{z}_i(1-\tilde{z}_i) - (1-\kappa) z_+ z_- \\ - \frac{\kappa\mu_Q^2}{2\mu_Q^2 + (1 - 2\mu_Q^2 - \mu_k^2) y_{ij,k}} \Bigg] \Bigg\},\end{aligned} \tag{A.2}$$

where we neglect the last term because of $\kappa = 0$ in our implementation. The integration ranges are

$$\frac{2\mu_j^2}{1 - 2\mu_j^2 - \mu_k^2} \ < y_{ij,k} \ < \ y_+ = 1 - \frac{2\mu_k(1-\mu_k)}{1 - 2\mu_j^2 - \mu_k^2}, \tag{A.3}$$

$$\frac{1 - v_{ij,i} v_{ij,k}}{2} \ < \tilde{z}_i \ < \frac{1 + v_{ij,i} v_{ij,k}}{2}. \tag{A.4}$$

Here, the integration over $y_{ij,k}$ does not start at $y_{ij,k} = 0$ but at a value larger than zero. But introducing the α parameter and calculating the correction terms implies new integration boundaries for the $y_{ij,k}$ integration according to (A.1):

$$\frac{2\mu_j^2}{1-2\mu_j^2-\mu_k^2} \;<\; \alpha y_+, \tag{A.5}$$

which implies that α must not be chosen to be too small for this splitting. With this restriction, we find

$$\begin{aligned}
I_{ij,k}^{(\mathrm{g})}(\epsilon,\alpha) = I_{ij,k}^{(\mathrm{g})}(\epsilon) - T_{\mathrm{R}} \Big(\Big(& bd \left(-8a\mu_j^4 \log\left(\alpha c^2 y_+ - d\sqrt{c^2-4\mu_j^4} - 4\mu_j^4\right) + 2ac^2 \log\left(\alpha c^2 y_+ - d\sqrt{c^2-4\mu_j^4} - 4\mu_j^4\right)\right. \\
& +2ac \log\left(\alpha c^2 y_+ - d\sqrt{c^2-4\mu_j^4} - 4\mu_j^4\right) + 4a\mu_j^2 \log\left(\alpha c^2 y_+ - d\sqrt{c^2-4\mu_j^4} - 4\mu_j^4\right) \\
& -2a\left(c^2+c-4\mu_j^4+2\mu_j^2\right) \log\left((1-\alpha y_+)\left(-c-2\mu_j^2\right)^{5/2}\left(c-2\mu_j^2+1\right)\right) \\
& +8a\mu_j^4 \log\left(-b\sqrt{c^2-4\mu_j^4} + c^2 y_+ - 4\mu_j^4\right) - 2ac^2 \log\left(-b\sqrt{c^2-4\mu_j^4} + c^2 y_+ - 4\mu_j^4\right) \\
& -2ac \log\left(-b\sqrt{c^2-4\mu_j^4} + c^2 y_+ - 4\mu_j^4\right) - 4a\mu_j^2 \log\left(-b\sqrt{c^2-4\mu_j^4} + c^2 y_+ - 4\mu_j^4\right) \\
& +2a\left(c^2+c-4\mu_j^4+2\mu_j^2\right) \log\left((y_+-1)\left(-\left(-c-2\mu_j^2\right)^{5/2}\right)\left(c-2\mu_j^2+1\right)\right) \\
& -3c^2\sqrt{2\mu_j^2-c}\log(-2(\alpha c y_+ + d)) + 4c\mu_j^2\sqrt{2\mu_j^2-c}\log(-2(\alpha c y_+ + d)) \\
& -2c\sqrt{2\mu_j^2-c}\log(-2(\alpha c y_+ + d)) + 3c^2\sqrt{2\mu_j^2-c}\log(-2(b+cy_+)) \\
& \left. -4c\mu_j^2\sqrt{2\mu_j^2-c}\log(-2(b+cy_+)) + 2c\sqrt{2\mu_j^2-c}\log(-2(b+cy_+))\right) \\
& +2bd\sqrt{2\mu_j^2-c}\left(c^2-2(c+1)\mu_j^2+4\mu_j^4\right)\tan^{-1}\left(\frac{2\mu_j^2}{\sqrt{\alpha^2c^2y_+^2-4\mu_j^4}}\right) \\
& +c\sqrt{2\mu_j^2-c}\left(c^2 y_+\left(\alpha^2 b y_+ - 2\alpha b - d(y_+-2)\right) + 4c\mu_j^2(b(\alpha y_+-1)+d(-y_+)+d) + 4\mu_j^4(b-d)\right) \\
& \left. -2bd\sqrt{2\mu_j^2-c}\left(c^2-2(c+1)\mu_j^2+4\mu_j^4\right)\tan^{-1}\left(\frac{2\mu_j^2}{\sqrt{c^2y_+^2-4\mu_j^4}}\right)\right) \\
& / \left(3c\left(2\mu_j^2-c\right)^{3/2}\sqrt{c^2y_+^2-4\mu_j^4}\sqrt{\alpha^2c^2y_+^2-4\mu_j^4}\right)\Big)
\end{aligned} \tag{A.6}$$

where we used the following abbreviations:

$$\begin{aligned}
a &= \sqrt{1-\mu_k^2} \\
b &= \sqrt{c^2y_+^2-4\mu_j^4} \\
c &= -1+2\mu_j^2+\mu_k^2 \\
d &= \sqrt{\alpha^2c^2y_+^2-4\mu_j^4}\,.
\end{aligned}$$

A.1.2 Case (h)

Case (h) describes also the splitting of a gluon into a massive quark pair ($g \to Q\,\bar{Q}$) however with a massless spectator. While the splitting function is the same (A.2), the integration

boundaries are different, namely

$$\frac{2\mu_i^2}{1-2\mu_i^2} < y_{ij,k} < 1 \tag{A.7}$$

$$\frac{1-v_{ij,i}}{2} < \tilde{z}_i < \frac{1+v_{ij,i}}{2}. \tag{A.8}$$

The phase space integral gets multiplied by $\Theta(\alpha - y_{ij,k})$ and integration leads to

$$\begin{aligned} I^{(\mathrm{h})}_{ij,k}(\epsilon,\alpha) = I^{(\mathrm{h})}_{ij,k}(\epsilon) - T_{\mathrm{R}} \Bigg(\frac{2}{3} \Bigg(& \frac{2\sqrt{\alpha^2\left(1-2\mu_j^2\right)^2 - 4\mu_j^4}}{2(\alpha-1)\mu_j^2 - \alpha} + \sqrt{\alpha^2\left(1-2\mu_j^2\right)^2 - 4\mu_j^4} + \left(2\mu_j^2 - 1\right) \\ & \left(-\log\left(-2\left(\sqrt{\alpha^2\left(1-2\mu_j^2\right)^2 - 4\mu_j^4} + \alpha\left(2\mu_j^2-1\right)\right)\right) + 2\tan^{-1}\left(\frac{2\mu_j^2}{\sqrt{\alpha^2\left(1-2\mu_j^2\right)^2 - 4\mu_j^4}}\right)\right. \\ & \left. + \log\left(-2\left(2\mu_j^2 + \sqrt{1-4\mu_j^2} - 1\right)\right) - 2\tan^{-1}\left(\frac{2\mu_j^2}{\sqrt{1-4\mu_j^2}}\right)\right) + \sqrt{1-4\mu_j^2}\Bigg)\Bigg). \end{aligned} \tag{A.9}$$

A.2 Final-initial

Configuration	Reference	Remarks
(a) $Q \to Q\ g$	[64] (A.13,A.14)	
(b) $q \to q\ g$	[64](A.17) / [13] (11-16)	Different approaches
(c) $g \to Q\ \bar{Q}$	this work, (A.12,A.14,A.16)	
(d) $g \to q\ \bar{q}$	[13] (11-16)	Different approach in MadDipole
(e) $g \to g\ g$	[13] (11-16)	Different approach in MadDipole

The cases (b), (d), and (e) can be found in [13]. Their result however contains already the sum of different contributions making use of the I- and K-flavor kernels. As it mixes different contributions this is not suitable for the MadDipole implementation. Therefore we use [64] for (b) and derive results for (d) and (e) following the approach in [64]. Of course, both approaches are equivalent. Again, for the sake of completeness we also add the finite case (c).

A.2.1 Case (c)

The one particle phase space for final-initial dipoles is given in Eq.(5.48) of [50]:

$$\begin{aligned} \int [\mathrm{d}p_i(\widetilde{p}_{ij}; p_a, x)] = & \frac{1}{4}(2\pi)^{-3+2\epsilon}(2\widetilde{p}_{ij}p_a)^{1-\epsilon} \int_0^{x_+} \mathrm{d}x_{ij,a}\, \delta(x - x_{ij,a})\, (1 - x + \mu_{ij}^2)^{-\epsilon} \\ & \times \int \mathrm{d}^{d-3}\Omega \int_{z_-(x)}^{z_+(x)} \mathrm{d}\tilde{z}_i\, \left[z_+(x) - \tilde{z}_i\right]^{-\epsilon} \left[\tilde{z}_i - z_-(x)\right]^{-\epsilon}, \end{aligned} \tag{A.10}$$

and the integrated splitting function is given by Eq.(5.53) of [50] as:

$$\int [\mathrm{d}p_i(\widetilde{p}_{ij}; p_a, x)]\, \frac{1}{(p_i+p_j)^2 - m_{ij}^2} \langle \mathbf{V}_{ij}^a \rangle \equiv \frac{\alpha_{\mathrm{s}}}{2\pi} \frac{1}{\Gamma(1-\epsilon)} \left(\frac{4\pi\mu^2}{2\widetilde{p}_{ij}p_a} \right)^{\epsilon} I_{ij}^a(x;\epsilon)\,. \tag{A.11}$$

For case (c), the splitting of a gluon into massive quarks ($g \to Q\,\bar{Q}$), the integrated spitting function is given in Eq.(5.57) of [50]:

$$I^a_{Q\bar{Q}}(x;\epsilon) = T_{\rm R}\left\{[J^a_{Q\bar{Q}}(x,\mu_Q)]_{x_+} + \delta(x_+ - x)\left[J^{a;{\rm S}}_{Q\bar{Q}}(\mu_Q;\epsilon) + J^{a;{\rm NS}}_{Q\bar{Q}}(\mu_Q)\right]\right\} + {\rm O}(\epsilon)\,, \quad \text{(A.12)}$$

where the x_+-distribution is defined as:

$$\int_0^1 {\rm d}x \left(f(x)\right)_{x_+} g(x) \equiv \int_0^1 {\rm d}x\, f(x)\Theta(x_+ - x)\,[g(x) - g(x_+)]\,. \quad \text{(A.13)}$$

Imposing the cut on the α-parameter implies that the phase space in (A.10) is multiplied with $\Theta(\alpha - 1 + x_{ija})$.

This leads to a modification of the x_+-distribution terms and we get in analogy to Eq.(5.62) of [50]

$$[J^a_{Q\bar{Q}}(x,\mu_Q,\alpha)]_{x_+} = \frac{2}{3}\left(\frac{1-x+2\mu_Q^2}{(1-x)^2}\sqrt{1-\frac{4\mu_Q^2}{1-x}}\right)_{1-\alpha}\,, \quad \text{(A.14)}$$

where we define:

$$\int\limits_0^1 dx\ f(x)\,(g(x))_{1-\alpha} = \int\limits_{1-\alpha}^1 dx\ g(x)\,(f(x) - f(1)) \quad \text{(A.15)}$$

The non-singular terms $J^{a;{\rm NS}}_{Q\bar{Q}}(\mu_Q)$ receive:

$$J^{a;{\rm NS}}_{Q\bar{Q}}(\mu_Q,\alpha) = J^{a;{\rm NS}}_{Q\bar{Q}}(\mu_Q) + \frac{2}{9}\left(\left(-4\mu_Q^2\left(\sqrt{\frac{\left(1-4\mu_Q^2\right)\left(\alpha-4\mu_Q^2\right)}{\alpha^3}}+4\right) - 5\sqrt{\frac{\left(1-4\mu_Q^2\right)\left(\alpha-4\mu_Q^2\right)}{\alpha}}\right.\right.$$
$$\left.\left. -16\mu_Q^4 + 5\right) / \left(\sqrt{1-4\mu_Q^2}\right) + 6\log\left(\sqrt{\alpha-4\mu_Q^2}+\sqrt{\alpha}\right) - 6\log\left(\sqrt{1-4\mu_Q^2}+1\right)\right)\,. \quad \text{(A.16)}$$

A.2.2 Case (d)

Case (d) is just the limit $\mu_Q \to 0$ of (A.14), *i.e.*,

$$[J^a_{q\bar{q}}(x,0,\alpha)]_+ = \frac{2}{3}\left(\frac{1}{1-x}\right)_{1-\alpha}\,, \quad \text{(A.17)}$$

which leads to the following additional non-singular terms:

$$J^{a;{\rm NS}}_{q\bar{q}}(0,\alpha) = J^{a;{\rm NS}}_{q\bar{q}}(0) + \frac{2}{3}\log\alpha. \quad \text{(A.18)}$$

A.2.3 Case (e)

In the case of the splitting ($g \to g\ g$) the general structure of the integrated splitting function is given by Eq.(5.66) of [50]:

$$I^a_{gg}(x;\epsilon) = 2C_{\rm A}\left\{[J^a_{gg}(x)]_+ + \delta(1-x)\,J^{a;{\rm S}}_{gg}(\epsilon)\right\} + {\rm O}(\epsilon)\,. \quad \text{(A.19)}$$

The first term $[J^a_{gg}(x)]_+$ contains all +-distributions but is not a +-distribution itself. In the presence of the α-parameter we find

$$[J^a_{gg}(x,\alpha)]_+ = \left(\frac{2}{1-x}\ln\frac{1}{1-x} - \frac{11}{6}\frac{1}{1-x}\right)_{1-\alpha} + \frac{2}{1-x}\ln(2-x)\Theta(\alpha-1+x)\,, \quad \text{(A.20)}$$

which leads to a modification of the terms proportional to $\delta(1-x)$ of Eq.(5.68) of [50] of the following form:

$$J^{a;\mathrm{S}}_{gg}(x,\alpha) = J^{a;\mathrm{S}}_{gg}(x) - \log^2\alpha - \frac{11}{6}\log\alpha. \quad \text{(A.21)}$$

A.3 Initial-final

	Configuration		Reference	Remarks
(a)	$\tilde{ij}:\ q,$	emitter : q	[64] (A.9, A.11)/ [13] (11-16)	[13] only massless
(b)	$\tilde{ij}:\ g,$	emitter : q	[65] (A.8)	
(c)	$\tilde{ij}:\ q,$	emitter : g	[65] (A.10)	
(d)	$\tilde{ij}:\ g,$	emitter : g	[65] (A.11)	

From the analytical point of view the limit of a vanishing spectator mass can be performed without any problems. However taking the result for a massive spectator and setting the mass to zero in the numerical implementation causes problems for the cases (b) and (d). For these two cases we calculated the limit analytically and implemented a massive and a massless version.

A.4 Initial-initial

	Configuration		Reference	Remarks
(a)	$\tilde{ij}:\ q,$	emitter : q	[64] (A.4)/ [13] (11-16)	
(b)	$\tilde{ij}:\ g,$	emitter : q	[64] (A.5)/ [13] (11-16)	
(c)	$\tilde{ij}:\ q,$	emitter : g	[65] (A.4)/ [13] (11-16)	
(d)	$\tilde{ij}:\ g,$	emitter : g	[65] (A.5)/ [13] (11-16)	

For our implementation the results from [64, 65] are used.

References

[1] T. Stelzer and W. F. Long, "Automatic generation of tree level helicity amplitudes," *Comput. Phys. Commun.* **81** (1994) 357–371, `hep-ph/9401258`.

[2] F. Maltoni and T. Stelzer, "MadEvent: Automatic event generation with MadGraph," *JHEP* **02** (2003) 027, `hep-ph/0208156`.

[3] J. Alwall *et al.*, "MadGraph/MadEvent v4: The New Web Generation," *JHEP* **09** (2007) 028, `0706.2334`.

[4] **CompHEP** Collaboration, E. Boos *et al.*, "CompHEP 4.4: Automatic computations from Lagrangians to events," *Nucl. Instrum. Meth.* **A534** (2004) 250–259, `hep-ph/0403113`.

[5] A. Pukhov, "CalcHEP 3.2: MSSM, structure functions, event generation, batchs, and generation of matrix elements for other packages," hep-ph/0412191.

[6] T. Gleisberg *et al.*, "SHERPA 1.α, a proof-of-concept version," *JHEP* **02** (2004) 056, hep-ph/0311263.

[7] T. Gleisberg *et al.*, "Event generation with SHERPA 1.1," *JHEP* **02** (2009) 007, 0811.4622.

[8] W. Kilian, T. Ohl, and J. Reuter, "WHIZARD: Simulating Multi-Particle Processes at LHC and ILC," 0708.4233.

[9] M. L. Mangano, M. Moretti, F. Piccinini, R. Pittau, and A. D. Polosa, "ALPGEN, a generator for hard multiparton processes in hadronic collisions," *JHEP* **07** (2003) 001, hep-ph/0206293.

[10] A. Cafarella, C. G. Papadopoulos, and M. Worek, "Helac-Phegas: a generator for all parton level processes," *Comput. Phys. Commun.* **180** (2009) 1941–1955, 0710.2427.

[11] J. M. Campbell and R. K. Ellis, "An update on vector boson pair production at hadron colliders," *Phys. Rev.* **D60** (1999) 113006, hep-ph/9905386.

[12] J. Campbell and R. K. Ellis, "Next-to-leading order corrections to W + 2jet and Z + 2jet production at hadron colliders," *Phys. Rev.* **D65** (2002) 113007, hep-ph/0202176.

[13] Z. Nagy, "Next-to-leading order calculation of three-jet observables in hadron hadron collision," *Phys. Rev.* **D68** (2003) 094002, hep-ph/0307268.

[14] S. Frixione and B. R. Webber, "Matching NLO QCD computations and parton shower simulations," *JHEP* **06** (2002) 029, hep-ph/0204244.

[15] S. Frixione, P. Nason, and B. R. Webber, "Matching NLO QCD and parton showers in heavy flavour production," *JHEP* **08** (2003) 007, hep-ph/0305252.

[16] S. Frixione, E. Laenen, P. Motylinski, and B. R. Webber, "Single-top production in MC@NLO," *JHEP* **03** (2006) 092, hep-ph/0512250.

[17] P. Torrielli and S. Frixione, "Matching NLO QCD computations with PYTHIA using MC@NLO," 1002.4293.

[18] P. Nason, "A new method for combining NLO QCD with shower Monte Carlo algorithms," *JHEP* **11** (2004) 040, hep-ph/0409146.

[19] P. Nason and G. Ridolfi, "A positive-weight next-to-leading-order Monte Carlo for Z pair hadroproduction," *JHEP* **08** (2006) 077, hep-ph/0606275.

[20] O. Latunde-Dada, S. Gieseke, and B. Webber, "A positive-weight next-to-leading-order Monte Carlo for e^+e^- annihilation to hadrons," *JHEP* **02** (2007) 051, hep-ph/0612281.

[21] S. Frixione, P. Nason, and G. Ridolfi, "A Positive-Weight Next-to-Leading-Order Monte Carlo for Heavy Flavour Hadroproduction," *JHEP* **09** (2007) 126, 0707.3088.

[22] S. Alioli, P. Nason, C. Oleari, and E. Re, "NLO vector-boson production matched with shower in POWHEG," *JHEP* **07** (2008) 060, 0805.4802.

[23] K. Hamilton, P. Richardson, and J. Tully, "A Positive-Weight Next-to-Leading Order Monte Carlo Simulation of Drell-Yan Vector Boson Production," 0806.0290.

[24] S. Alioli, P. Nason, C. Oleari, and E. Re, "A general framework for implementing NLO calculations in shower Monte Carlo programs: the POWHEG BOX," *JHEP* **06** (2010) 043, 1002.2581.

[25] D. B. Melrose, "Reduction of Feynman diagrams," *Nuovo Cim.* **40** (1965) 181–213.

[26] G. Passarino and M. J. G. Veltman, "One Loop Corrections for e+ e- Annihilation Into mu+ mu- in the Weinberg Model," *Nucl. Phys.* **B160** (1979) 151.

[27] Z. Bern, L. J. Dixon, and D. A. Kosower, "Dimensionally regulated pentagon integrals," *Nucl. Phys.* **B412** (1994) 751–816, hep-ph/9306240.

[28] G. Ossola, C. G. Papadopoulos, and R. Pittau, "CutTools: a program implementing the OPP reduction method to compute one-loop amplitudes," *JHEP* **03** (2008) 042, 0711.3596.

[29] G. Ossola, C. G. Papadopoulos, and R. Pittau, "On the Rational Terms of the one-loop amplitudes," *JHEP* **05** (2008) 004, 0802.1876.

[30] C. F. Berger *et al.*, "An Automated Implementation of On-Shell Methods for One- Loop Amplitudes," *Phys. Rev.* **D78** (2008) 036003, 0803.4180.

[31] W. T. Giele and G. Zanderighi, "On the Numerical Evaluation of One-Loop Amplitudes: The Gluonic Case," *JHEP* **06** (2008) 038, 0805.2152.

[32] T. Binoth, J. P. Guillet, G. Heinrich, E. Pilon, and T. Reiter, "Golem95: a numerical program to calculate one-loop tensor integrals with up to six external legs," *Comput. Phys. Commun.* **180** (2009) 2317–2330, 0810.0992.

[33] A. Denner and S. Dittmaier, "Reduction schemes for one-loop tensor integrals," *Nucl. Phys.* **B734** (2006) 62–115, hep-ph/0509141.

[34] A. Denner, S. Dittmaier, M. Roth, and L. H. Wieders, "Electroweak corrections to charged-current $e^+ e^- \to 4$ fermion processes: Technical details and further results," *Nucl. Phys.* **B724** (2005) 247–294, hep-ph/0505042.

[35] C. F. Berger *et al.*, "Precise Predictions for W + 3 Jet Production at Hadron Colliders," *Phys. Rev. Lett.* **102** (2009) 222001, 0902.2760.

[36] C. F. Berger *et al.*, "Next-to-Leading Order QCD Predictions for W+3-Jet Distributions at Hadron Colliders," *Phys. Rev.* **D80** (2009) 074036, 0907.1984.

[37] R. K. Ellis, K. Melnikov, and G. Zanderighi, "Generalized unitarity at work: first NLO QCD results for hadronic W+ 3jet production," *JHEP* **04** (2009) 077, 0901.4101.

[38] R. Keith Ellis, K. Melnikov, and G. Zanderighi, "W+3 jet production at the Tevatron," *Phys. Rev.* **D80** (2009) 094002, 0906.1445.

[39] C. F. Berger *et al.*, "Next-to-Leading Order QCD Predictions for Z, γ^*+3-Jet Distributions at the Tevatron," 1004.1659.

[40] A. Bredenstein, A. Denner, S. Dittmaier, and S. Pozzorini, "NLO QCD corrections to pp $\to$ t anti-t b anti-b + X at the LHC," *Phys. Rev. Lett.* **103** (2009) 012002, 0905.0110.

[41] A. Bredenstein, A. Denner, S. Dittmaier, and S. Pozzorini, "NLO QCD corrections to top anti-top bottom anti-bottom production at the LHC: 2. full hadronic results," *JHEP* **03** (2010) 021, 1001.4006.

[42] G. Bevilacqua, M. Czakon, C. G. Papadopoulos, R. Pittau, and M. Worek, "Assault on the NLO Wishlist: pp $\to$ tt bb," *JHEP* **09** (2009) 109, 0907.4723.

[43] G. Bevilacqua, M. Czakon, C. G. Papadopoulos, and M. Worek, "Dominant QCD Backgrounds in Higgs Boson Analyses at the LHC: A Study of pp $\to$ t anti-t + 2 jets at Next-To-Leading Order," 1002.4009.

[44] T. Binoth *et al.*, "Next-to-leading order QCD corrections to $pp \to b\bar{b}b\bar{b} + X$ at the LHC: the quark induced case," *Phys. Lett.* **B685** (2010) 293–296, 0910.4379.

[45] W. T. Giele and E. W. N. Glover, "Higher order corrections to jet cross-sections in e^+ e^- annihilation," *Phys. Rev.* **D46** (1992) 1980–2010.

[46] Z. Kunszt and D. E. Soper, "Calculation of jet cross-sections in hadron collisions at order α_s^3," *Phys. Rev.* **D46** (1992) 192–221.

[47] S. Frixione, Z. Kunszt, and A. Signer, "Three-jet cross sections to next-to-leading order," *Nucl. Phys.* **B467** (1996) 399–442, hep-ph/9512328.

[48] G. Somogyi, "Subtraction with hadronic initial states: an NNLO- compatible scheme," *JHEP* **05** (2009) 016, 0903.1218.

[49] S. Catani and M. H. Seymour, "A general algorithm for calculating jet cross sections in NLO QCD," *Nucl. Phys.* **B485** (1997) 291–419, hep-ph/9605323.

[50] S. Catani, S. Dittmaier, M. H. Seymour, and Z. Trocsanyi, "The dipole formalism for next-to-leading order QCD calculations with massive partons," *Nucl. Phys.* **B627** (2002) 189–265, hep-ph/0201036.

[51] D. A. Kosower, "Antenna factorization of gauge-theory amplitudes," *Phys. Rev.* **D57** (1998) 5410–5416, hep-ph/9710213.

[52] J. M. Campbell, M. A. Cullen, and E. W. N. Glover, "Four jet event shapes in electron positron annihilation," *Eur. Phys. J.* **C9** (1999) 245–265, hep-ph/9809429.

[53] A. Gehrmann-De Ridder, T. Gehrmann, and E. W. N. Glover, "Antenna subtraction at NNLO," *JHEP* **09** (2005) 056, hep-ph/0505111.

[54] A. Daleo, T. Gehrmann, and D. Maître, "Antenna subtraction with hadronic initial states," *JHEP* **04** (2007) 016, hep-ph/0612257.

[55] T. Gleisberg and F. Krauss, "Automating dipole subtraction for QCD NLO calculations," *Eur. Phys. J.* **C53** (2008) 501–523, 0709.2881.

[56] M. H. Seymour and C. Tevlin, "TeVJet: A general framework for the calculation of jet observables in NLO QCD," 0803.2231.

[57] M. Czakon, C. G. Papadopoulos, and M. Worek, "Polarizing the Dipoles," *JHEP* **08** (2009) 085, 0905.0883.

[58] K. Hasegawa, S. Moch, and P. Uwer, "AutoDipole – Automated generation of dipole subtraction terms –," 0911.4371.

[59] R. Frederix, T. Gehrmann, and N. Greiner, "Automation of the Dipole Subtraction Method in MadGraph/MadEvent," *JHEP* **09** (2008) 122, 0808.2128.

[60] R. Frederix, S. Frixione, F. Maltoni, and T. Stelzer, "Automation of next-to-leading order computations in QCD: the FKS subtraction," *JHEP* **10** (2009) 003, 0908.4272.

[61] **SM and NLO Multileg Working Group** Collaboration, J. R. Andersen *et al.*, "The SM and NLO multileg working group: Summary report," 1003.1241.

[62] T. Binoth *et al.*, "A proposal for a standard interface between Monte Carlo tools and one-loop programs," 1001.1307.

[63] Z. Nagy and Z. Trocsanyi, "Next-to-leading order calculation of four-jet observables in electron positron annihilation," *Phys. Rev.* **D59** (1999) 014020, hep-ph/9806317.

[64] J. Campbell, R. K. Ellis, and F. Tramontano, "Single top production and decay at next-to-leading order," *Phys. Rev.* **D70** (2004) 094012, hep-ph/0408158.

[65] J. Campbell and F. Tramontano, "Next-to-leading order corrections to W t production and decay," *Nucl. Phys.* **B726** (2005) 109–130, hep-ph/0506289.

[66] G. 't Hooft and M. J. G. Veltman, "Regularization and Renormalization of Gauge Fields," *Nucl. Phys.* **B44** (1972) 189–213.

[67] C. G. Bollini and J. J. Giambiagi, "Dimensional Renormalization: The Number of Dimensions as a Regularizing Parameter," *Nuovo Cim.* **B12** (1972) 20–25.

[68] J. F. Ashmore, "A Method of Gauge Invariant Regularization," *Lett. Nuovo Cim.* **4** (1972) 289–290.

[69] G. M. Cicuta and E. Montaldi, "Analytic renormalization via continuous space dimension," *Nuovo Cim. Lett.* **4** (1972) 329–332.

[70] W. Siegel, "Supersymmetric Dimensional Regularization via Dimensional Reduction," *Phys. Lett.* **B84** (1979) 193.

[71] W. Siegel, "Inconsistency of Supersymmetric Dimensional Regularization," *Phys. Lett.* **B94** (1980) 37.

[72] D. Stockinger, "Regularization by dimensional reduction: Consistency, quantum action principle, and supersymmetry," *JHEP* **03** (2005) 076, hep-ph/0503129.

[73] A. Signer and D. Stockinger, "Using Dimensional Reduction for Hadronic Collisions," *Nucl. Phys.* **B808** (2009) 88–120, 0807.4424.

[74] Z. Kunszt, A. Signer, and Z. Trocsanyi, "One loop helicity amplitudes for all $2 \to 2$ processes in QCD and N=1 supersymmetric Yang-Mills theory," *Nucl. Phys.* **B411** (1994) 397–442, hep-ph/9305239.

[75] S. Catani, S. Dittmaier, and Z. Trocsanyi, "One-loop singular behaviour of QCD and SUSY QCD amplitudes with massive partons," *Phys. Lett.* **B500** (2001) 149–160, hep-ph/0011222.